CLAIMS

What is claimed is:

1. 1. A method for equipping a simulator with plug-ins, comprising the steps of:
2. (a) executing a first simulator for generating a first model, wherein the first
3. simulator is written in a first programming language;
4. (b) executing a second simulator for generating a second model, wherein the
5. second simulator is written in a second programming language, and the first
6. simulator interfaces with the second simulator via a plug-in; and
7. (c) co-simulating utilizing the first model and the second model.

1. 2. A method as recited in claim 1, wherein an accuracy and speed of the co-
2. simulation is user-specified.

1. 3. A method as recited in claim 1, wherein the first simulator is cycle-based and
2. the second simulator is event-based.

1. 4. A method as recited in claim 1, wherein the co-simulation includes
2. interleaved scheduling.

1. 5. A method as recited in claim 1, wherein the co-simulation includes fully
2. propagated scheduling.

1. 6. A method as recited in claim 1, wherein the simulations are executed
2. utilizing a plurality of processors.

1. 7. A method as recited in claim 1, wherein the first simulator may be executed
2. ahead of or behind the second simulator.

1   8.   A method as recited in claim 1, wherein the first simulator is coupled to the

2   second simulator via a network.

1   9.   A computer program product for equipping a simulator with plug-ins,

2   comprising:

3   (a)   computer code for executing a first simulator for generating a first model,

4   wherein the first simulator is written in a first programming language;

5   (b)   computer code for executing a second simulator for generating a second

6   model, wherein the second simulator is written in a second programming

7   language, and the first simulator interfaces with the second simulator via a

8   plug-in; and

9   (c)   computer code for co-simulating utilizing the first model and the second

10   model.

1   10.   A computer program product as recited in claim 9, wherein an accuracy and

2   speed of the co-simulation is user-specified.

1   11.   A computer program product as recited in claim 9, wherein the first

2   simulator is cycle-based and the second simulator is event-based.

1   12.   A computer program product as recited in claim 9, wherein the co-simulation

2   includes interleaved scheduling.

1   13.   A computer program product as recited in claim 9, wherein the co-simulation

2   includes fully propagated scheduling.

1   14.   A computer program product as recited in claim 9, wherein the simulations

2   are executed utilizing a plurality of processors.

1   15.   A computer program product as recited in claim 9, wherein the first

2   simulator may be executed ahead of or behind the second simulator.

1 16. A computer program product as recited in claim 9, wherein the first
2 simulator is coupled to the second simulator via a network.

1 17. A system for equipping a simulator with plug-ins, comprising:
2 (a) logic for executing a first simulator for generating a first model, wherein the
3 first simulator is written in a first programming language;
4 (b) logic for executing a second simulator for generating a second model,
5 wherein the second simulator is written in a second programming language,
6 and the first simulator interfaces with the second simulator via a plug-in; and
7 (c) logic for co-simulating utilizing the first model and the second model.

1 18. A system as recited in claim 17, wherein an accuracy and speed of the co-
2 simulation is user-specified.

1 19. A system as recited in claim 17, wherein the first simulator is cycle-based
2 and the second simulator is event-based.

1 20. A system as recited in claim 17, wherein the co-simulation includes
2 interleaved scheduling.